

Projekt Java

Jakub Stachurski

January 8, 2024

1 Temat Projektu i jego autorzy

Jakub Stachurski ZADANIE 5 2020 - " MNOŻENIE BEZ KALKULATORA " Napisz program, który dla zadanych dwóch liczb zmiennoprzecinkowych zwróci iloczyn tych liczb. UWAGA: program (oczywiście) nie może korzystać z mnożenia, nie może również używać innych (poza dodawaniem) funkcji matematycznych (np. mnożenie, dzielenie, potęga, pierwiastek, część całkowita, zaokrąglenie, operacja modulo, itd.).

2 Opis pobieranych danych

Program przyjmuje dwie liczby zmiennoprzecinkowe, podane przez użytkownika w oknie konsoli. Dodatkowo musi on przestrzegać formatu gdzie należy podać minimum jedna cyfre, jeden przecinek i minimum jedna cyfre po przecinka. W innym wypadku program nie przyjmie wartości wpisanej przez użytkownika.

3 Opis otrzymanych rezultatów

Otrzymujemy wynik "mnożenia" dwóch liczb zmiennoprzecinkowych, w całym programie wykorzystując jedynie jedną operację matematyczną, jaką jest dodawanie. Tyczy się to również petli i instrukcji warunkowych.

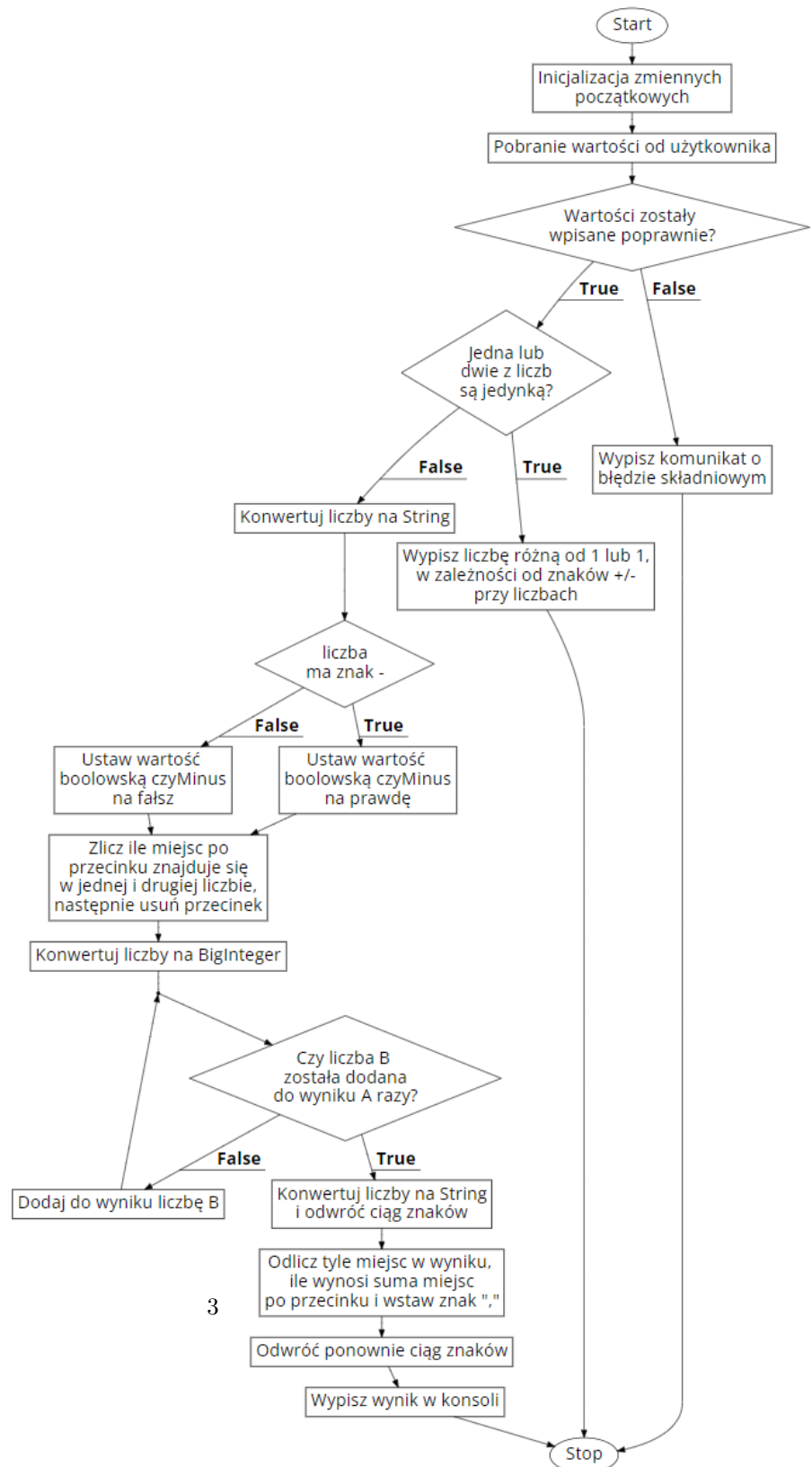
4 Zastosowany algorytm do rozwiązania zadania

4.1 Słowny opis

Program pobiera dwie liczby od użytkownika. Konwertuje liczby na typ String i sprawdza czy liczby są ujemne. Program zapamiętuje i ile minusów był i czy należy w ostatecznym wyniku dopisać znak "-". Następnie program zlicza ile cyfr znajduje się przed przecinkiem i usuwa go. Następuje konwersja na BigInteger, po czym liczba A jest dodawana do siebie B razy. W kolejnym kroku ponownie następuje konwersja na Stringa i odwrócenie ciągu znaków. Ilość miejsc po przecinku jest odliczana i w odpowiednim miejscu dostawiany

jest przecinek. Ciąg znaków ponownie jest odwracany i w odpowiedniej już formie wypisywany.

4.2 Schemat Blokowy



5 Testy na poprawność działania programu

By przetestować działanie algorytmu należy wymyślić własny przykład i obliczyć go zarówno ręcznie jak i programistycznie i porównać wyniki.

21,37 wartość A

4,2 Wartość B

Zmiana przecinka - 2137 (2 miejsca po przecinku)

Zmiana przecinka - 42 (1 miejsce po przecinku)

0 iteracja - 2137

1 iteracja - 4274

.

.

.

41 iteracja - 89754

odwrócenie liczby - 45798

3 miejsca od lewej strony

457,98

odwócenie liczby - 89,754

Wynik to 89,754

6 Wnioski

Moja metoda jest praktycznie odzwierciedleniem sposobu mnożenia liczb w słupku.

Początkowo pojawiały się błędy podczas mnożenia przez kolejne potęgi 10,0. Program został sprawdzony i zabezpieczony tak, aby błędy nie pojawiały się ponownie, a liczba wyświetlana była zawsze w odpowiednim formacie.

Dodatkowo program powinien móc obsłużyć dowolne ilości cyfr przed i po przecinku ze względu na użycie BigInteger i BigDecimal, jednocześnie im więcej cyfr zostało użytych, tym dłużej program będzie próbował obliczyć dane "mnożenie"